Towards AI that Can Solve Social Dilemmas

Alexander Peysakhovich*
Facebook AI Research

Adam Lerer*
Facebook AI Research

Abstract

Many scenarios involve a tension between individual interest and the interests of others. Such situations are called social dilemmas. Because of their ubiquity in economic and social interactions constructing agents that can solve social dilemmas is of prime importance to researchers interested in multi-agent systems. We discuss why social dilemmas are particularly difficult, propose a way to measure the 'success' of a strategy, and review recent work on using deep reinforcement learning to construct agents that can do well in both perfect and imperfect information bilateral social dilemmas.

Introduction

How can an agent construct a good strategies for an environment which involves another agent? An early answer to this question was given by (Brown 1951) who considered the idea of 'fictitious play' - an agent is going to play some game once with another agent, if they have access to the game beforehand and they can iterate the game repeatedly in their own mind (ie. during the training phase) and use the strategies they discovered when faced with a real partner (ie. during the test phase). This idea, also called 'self-play', has become an important part of the artificial intelligence toolkit. Self-play where agents try to maximize their own rewards can lead to superhuman performance in zero-sum games like Backgammon (Tesauro 1995), poker (Brown, Ganzfried, and Sandholm 2015), or Go (Silver et al. 2016; 2017) but can lead to bad outcomes in general-sum environments (Sandholm and Crites 1996; Fudenberg and Levine 1998; Lerer and Peysakhovich 2017; Foerster et al. 2017c; Leibo et al. 2017). Recent work has begun to study modified self-play methods to construct good strategies for social dilemmas. In this short note we will review some recent results in this field.

First, we need to determine what it means to do well in a social dilemma. The repeated Prisoner's Dilemma (rPD) is perhaps the most studied social dilemma and gives us a good starting point. In the rPD conditionally cooperative strategies such as Tit-for-Tat (Axelrod 2006) or Win-Stay-Lose-Shift (Nowak and Sigmund 1993) perform well because they reward cooperation today with cooperation tomorrow and so stabilize cooperation while avoiding exploitation. These strategies are studied so heavily because they have intuitively appealing properties. They are nice (begin by cooperating), are simple to explain to a partner, cooperate with cooperators, do not get exploited by defectors, are forgiving (eventually return to cooperation if it breaks down). Importantly, if one can commit to them, they create incentives for a partner to behave cooperatively. A natural desiderata then is to ask for agents in complex social dilemmas that maintain the good properties of these well known PD strategies.

There are several issues in extending these ideas to more complex settings. First, in Markov games 'cooperation' and 'defection' are no longer single acts, but rather sequences of choices (Leibo et al. 2017; Peysakhovich and Lerer 2017a; Lerer and Peysakhovich 2017; Foerster et al. 2017c; Littman 2001). Here agents that want to maintain cooperation within the confines of a single game have to 1) infer whether their partner is cooperating or not, and 2) know how to respond to both of these contingencies. The work we survey here tries to bring ideas from repeated game theory (Fudenberg and Maskin 1986; Dutta 1995; Littman and Stone 2005; De Cote and Littman 2012) to the one-shot setup. There are several issues to overcome: first, rather than maintaining good outcomes by threats of different behavior in *the next iteration* of the game, agents must behave intelligently *within a single game*; second, multiple strategies may be outcome equivalent (e.g. going left then up or up and then left in a grid world); third, function approximation may lead to noise in implementation. We would like to adapt the ideas from repeated game theory to construct strategies that are robust to these issues.

The first set of results we focus on construct conditional cooperators for fully observed games (Lerer and Peysakhovich 2017). The paper in question introduces approximate Markov Tit-for-Tat (amTFT) which applies modified self-play to learn two policies at training time: a fully cooperative policy and a 'safe' policy (we refer to this as defection)[1] which forms an equilibrium with lower payoffs than cooperation.

At test time, the amTFT agent is matched with a partner

*Equal contribution. Author order was determined via random.org.

[1] In the PD this action is 'defect'. However, in social dilemmas that occur naturally in economic situations, such a safe policy is the outside option of 'stop transacting with this agent.'

whose policy is unknown. At each time step the amTFT agent computes the gain from the action their partner actually chose compared to the one prescribed by the cooperative policy. This can be done either using a learned Q function or via policy rollouts. We refer to this as a per period debit. If the total debit is below a threshold the agent behaves according to the cooperative policy. If the debit at some time period is above the threshold, the agent switches to the defecting policy for k turns and then returns to cooperation. This k is computed such that the partner's gains (debit) are smaller than the losses they incur (k lost turns of cooperation). The threshold trades off robustness to noise and function approximation with allowing the amTFT agent to be slightly exploitable. It is shown both analytically and experimentally that amTFT can maintain cooperation and avoid being exploited in social dilemmas, including ones where agents learn from raw pixels.

Recent work has argued that TFT-like properties need not be hardwired and strategies can be trained from scratch. Foerster et al. modifies policy gradient to take into account that one's partner is a reactive (rather than static) agent. This method can construct cooperation maintaining strategies in several Markov games. This approach is computationally challenging and has no known theoretical guarantees, and it may construct strategies that are hard to explain (e.g. to a human partner). Despite these drawbacks we believe end-to-end training is a fruitful direction for future research and that explicit constructions like the ones we discuss here are a complement to, not a substitute for, end-to-end approaches.

An advantage of amTFT is that it requires no additional machinery beyond what is required by standard self-play, thus if deep RL can construct competitive agents in an environment such as Atari (Mnih et al. 2015) then we can also construct agents that solve social dilemmas in that environment. A disadvantage is that it requires full observability of a partner's action as well as a good model of the future consequences of a partner's action. Thus, it will not work in many POMDPs. amTFT's focus on future expected rewards as the result of an action can be replaced by consequentialism (Peysakhovich and Lerer 2017a): focusing on the reward stream that one actually obtains. Consequentialist conditionally cooperative (CCC) use self play to compute cooperate and defect strategies like amTFT. CCC uses rollouts of these strategies to compute a time-dependent payoff threshold, if the CCC agent's payoff at a period is below this threshold they defect, otherwise they cooperate. As long as a POMDP satisfies a technical conditions (reward ergodicity) CCC agents can maintain cooperation in the long-run.

CCC is much simpler to compute than amTFT and can perform just as well in some perfect information games. However, this is not always the case. Consider a situation where a partner tries to cheat (very obviously) but due to stochasticity in the environment fails to do so. amTFT would correctly mark this as a deviation from cooperation (because it focuses on the 'intention' behind an action) while CCC would not (because it only looks at consequences). In reality intention is usually somewhat observed (but not perfectly) while consequences are also noisy. This suggests that an important future direction towards constructing agents that solve social dilemmas is finding ways to combine intention and consequences efficiently.

We now describe in more technical detail the results we have surveyed here. Note that the experiments described here are not new, rather they are taken from the papers in question and presented in a summarized way to convey our main points. We point the interested reader back to the original papers for the full details.

Cooperation With Perfect Information

We begin with a generalization of Markov decision problems:

Definition 1 ((Shapley 1953)) *A (finite, 2-player) Markov game consists of*

- *A set of states $S = \{s_1, \dots, s_n\}$*
- *A set of actions for each player $\mathcal{A}^1 = \{a^1_1, \dots, a^1_k\}$, $\mathcal{A}^2 = \{a^2_1, \dots, a^2_k\}$*
- *A transition function $\tau : S \times A_1 \times A_2 \rightarrow \Delta(S)$ which tells us the probability distribution on the next state as a function of current state and actions*
- *A reward function for each player $R_i : S \times A^1 \times A^2 \rightarrow \mathbb{R}$ which tells us the utility that player gains from a state, action tuple*

We assume rewards are bounded above and below. Players can choose between policies which are maps from states to probability distributions on actions $\pi_i : S \rightarrow \Delta(\mathcal{A}_i)$. We denote by Π_i the set of all policies for a player.

Definition 2 *A value function for a player i inputs a state and a pair of policies $V^i(s, \pi^1, \pi^2)$ and gives the expected discounted reward to that player from starting in state s. We assume agents discount the future with rate δ which we subsume into the value function.*

We will be talking about strategic agents so we often refer to the concept of a best response:

Definition 3 *A policy for agent j denoted π_j is a best response starting at state s to a policy π_i if for any π'_j and any s' along the trajectory generated by these policies we have*

$$V^j(s', \pi^i, \pi^j) \geq V^j(s', \pi^i, \pi'^j).$$

We denote the set of such best responses as $BR^j(\pi^i, s)$. If π_j obeys the inequality above for any choice of state s we call it a perfect best response.

The set of stable states in a game is the set of equilibria. We call a policy for player 1 and a policy for player 2 a Nash equilibrium if they are best responses to each other. We call them a Markov perfect equilibrium if they are perfect best responses.

We are interested in a special set of policies:

Definition 4 *Cooperative Markov policies starting from state s (π^1_C, π^2_C) are those which, starting from state s, maximize*

$$V^1(s, \pi^1, \pi^2) + V^2(s, \pi^1, \pi^2).$$

We let the set of cooperative policies be denoted by $\Pi^C_i(c)$. Let the set of policies which are cooperative from any state be the set of perfectly cooperative policies.

A social dilemma is a game where there are no cooperative policies which form equilibria. In other words, if one player commits to play a cooperative policy at every state, there is a way for the other to exploit them and earn higher rewards. Note that in a social dilemma there may be policies which achieve the *payoffs* of cooperative policies because they cooperate on the trajectory of play and prevent exploitation by threatening non-cooperation on states which are never reached by the trajectory.

For the same situation the choice of state representation can affect whether a social dilemma is solvable or unsolvable. To make this more clear, let us consider the repeated Prisoner's Dilemma. In the simplest version rPD individuals are matched to play infinitely many rounds of a stage game in which each player chooses in each round either to give the other player a benefit b at a cost c to themselves (cooperate) or not (defect). When $b > c$ the highest total payoff is achieved when both individuals cooperate, however, each can do better in the short-run by defecting.

The rPD as described in words above can be written as a Markov game in many ways. For example, we can say that there is a single state and two actions per period. In this case, the rPD is an unsolvable social dilemma. This is because the only way to deter defection today is to affect the future payoffs of the defecting agent. With single state, this is impossible. On the other hand, if we model the rPD as a Markov game where the state is the outcome from last period, there are now policies which maintain cooperation and are an equilibrium. For any state representation can never be equilibria which cooperate at *every* state in the rPD because deterring defection today depends on being willing to withhold cooperation from defectors tomorrow and so policies that maintain cooperation at some states must defect at others.

The distinction made above is important because in many examples of interest the simplest choice of representation may not be one that makes the dilemma solvable. In particular, this implies that to play from raw pixels some memory is required, either in the form of an RNN (or similar) or a hardcoded summary statistic. Note that adding memory can create equilibrium policies which maintain cooperation. However, it does not remove equilibria in which both players which always defect. Thus, even with memory applying the self-play paradigm of 'learn a Nash equilibrium at training time and then play your half at test time' may still lead to defecting agents. It has been demonstrated several times that such defecting equilibria can be more robust attractors than cooperative equilibria.

amTFT bypasses this problem by doing the following. When paired with an actual partner the amTFT agent starts in a C phase. While in a C phase the agent behaves according to π^C. However, at each time step while in the C phase the amTFT agent looks at the actions a partner (called j) takes and computes

$$d = Q^j_{CC}(s, \pi^C_i(s), a_j) - Q^j_{CC}(s, \pi^C_i(s), \pi^j_C(s)).$$

If $d > 0$ the amTFT agent switches to a D phase for k periods which is computed such that the loss to the partner from k periods of π^D followed by mutual π^C is relative to both behaving according to π^C the whole time is greater than d. In other words, if a partner deviates today, they lose k periods of cooperation tomorrow.

In Lerer and Peysakhovich, the following analytical result is shown:

Theorem (Intuitive Version) 1 *If the game satisfies some technical conditions which generalize the notion of a Prisoner's Dilemma then if agent j's partner is an amTFT agent, the best response for agent j to play according to π^C_j during the C phase and π^D_j during the D phase. This means that if agents start in a C phase they cooperate forever. If agents start in a D phase they eventually return to cooperation and cooperate forever.*

amTFT is implemented using deep reinforcement learning. Importantly, during training time the amTFT agent has to find cooperative policy π^C and a defect policy π^D. These are found using a modified self-play procedure where the agent either controls both agents and reinforces at each time step on the agents' individual rewards (this is standard self-play and is used to find π^D) or on the joint reward (this finds the joint payoff maximizing policies π^C). In addition, d and k are computed by rollouts and to deal with issues of function approximation d is aggregated over multiple time steps of the game and the D phase begins only if the sum of d passes a threshold.

Cooperation Without Perfect Information

With imperfect information we can use the generalization of a POMDP to the multi-agent case. Here, we take the Markov game definition above and append the notion of observational states. Each player has a set of possible observations O_i and a function Ω_i which maps the state and actions at a given time period to an observation. When Ω_i is the identity for all players we get back a Markov game. Policies, instead of being able to condition on the state, must condition only on observations.

Note that here amTFT is not implementable since the action of a partner may not be perfectly observed. An ideal solution may be to construct a full posterior belief on actions using Bayesian methods. However, often such solutions are intractable. It is possible to construct a simple strategy for any game which satisfies a reward-ergodicity condition: for any pair of policies, there exists a limiting average rate of rewards which is independent of initial starting state. Let ρ_{CC} be the asymptotic rate under joint cooperation and ρ_{CD} be the asymptotic rate under the CCC agent cooperating and the other defecting. We can construct a consequentialist conditionally cooperative (CCC) agent who looks at their current average per period payoff and cooperates if this is above $\alpha\rho_{CC} + (1 - \alpha)\rho_{CD}$ and defects otherwise. This gives a theoretical result:

Theorem (Intuitive Version) 2 *If the game satisfies some technical conditions on the strategies then if a CCC agent is paired with a cooperator they are both guaranteed their cooperative payoffs and if a CCC agent is paired with a defector the defector is guaranteed at most the joint defect payoffs.*

In practice we can construct CCC agents using the same modified self-play as amTFT during training to compute π^C and π^D. Rollouts are used to compute per-period thresholds (Peysakhovich and Lerer 2017a). Note that the analytic results are asymptotic in nature and use the ergodicity condition heavily. To make the CCC strategy work well in finite time we need to use batches of rollouts and suggest using statistics other than the mean (e.g. quantiles) from these batches to construct thresholds. This allows the strategy to trade off flexibly between finite time false positives (assuming a partner is defecting when they are not) and false negatives (missing a defector). Note that this is a purely finite-time tradeoff - asymptotically the theoretical guarantees continue to hold. See the original paper for more details.

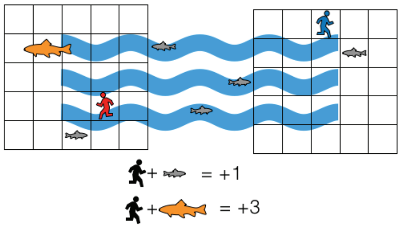
Experiments

We show the results of applying CCC and amTFT to several games. Here all results are trained using deep RL using standard methods. We refer the readers to the original papers for the full training details.

We also follow the metrics introduced in the original papers. We focus on the key desiderata: a good strategy should be safe from a defector partner, should incentivize cooperation from its partner, and, when matched with a conditional cooperator, should achieve good payoffs.

We define $S_i(X, Y)$ as the expected reward to policy π_1^X matched with π_2^Y. Safety$(X) = S_1(X, D) - S_1(D, D)$ measures how a strategy is safe from exploitation by a defector; and IncentC$(X) = S_2(X, C) - S_2(X, D)$ measures whether a strategy incentivizes cooperation from its partner. While we cannot enumerate all possible conditionally cooperative strategy, we can use a proxy in the case of CCC/amTFT. SelfMatch$(X) = S_1(X, X)$ measures whether a strategy achieves good outcomes with itself. We can compare this payoff to $S_1(C, C)$ and see how much cooperation these policies can achieve.

We begin with the results from using CCC in a POMDP: Fishery. Fishery is a grid-world partially observed Markov game where two agents live on 5×5 grids on opposite sides of a lake. Agents cannot observe the other side of the lake. Fish spawn in each agent's grid and start as young, if they are not caught when they are young they swim to the other side of the lake and become mature. Moving over a fish catches it. Catching a young fish is worth 1 point and catching a mature fish is worth 3 points. Thus, cooperative strategies are those which one catch mature fish but selfish agents are tempted to increase their payoff at a cost to their partner by catching young fish as well. Because this is a partially observed game, we can only use CCC as a cooperation maintaining strategy. We see that in this game agents that play the cooperative strategy (found by modified self-play with both agents receiving the joint reward at training time) can be exploited by defectors (this strategy is found by standard self-play). However CCC achieve cooperation with other CCC agents, is safe, and can incentivize its partner to cooperate.



Strategy	SelfMatch	Safety	IncentC
π^C	141	-36	-31
π^D	64	0	-34
CCC	125	-3	64

Figure 1: Fishery is a partially observed Markov social dilemma. Mutual cooperation leads to high payoffs but cooperators can be exploited by defectors. CCC cooperates with cooperators, is not exploited by defectors, and makes cooperation a high payoff strategy for its partner.

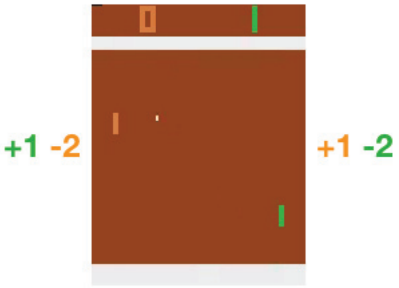
We now show the results of applying amTFT and CCC to a social dilemma where agents are trained directly from raw pixels. We can change the payoffs of Atari Pong to make it a social dilemma - the Pong Player's Dilemma or PPD (Tampuu et al. 2017). In the PPD when a player scores they receive a reward of 1 while their partner receives a reward of -2. Thus, cooperative strategies are those which gently hit the ball back and forth until the end of the game (and are exploitable by defectors who try hard to score). We see in Figure 2 that both amTFT and CCC perform well in the PPD - cooperating with cooperators, not being exploited by defectors, earning high payoffs when matched with other conditionally cooperative strategies, and incentivizing cooperation from a partner who can choose a strategy.

Because CCC is computationally simpler, one may believe the last result implies it is strictly better than amTFT. This is not always the case. We can change the payoff structure of the PPD to make it stochastic – when a player scores a point their partner gets a reward of $-\frac{2}{p}$ with probability p. We call this the risky PPD. Thus, the expected reward is the same as in the PPD but if p is low then most of the time the cooperative and defect trajectories look identical from the point of view of the payoffs. Here, CCC can be exploited by a defector while amTFT (which uses expected future payoffs) behaves the same as in the standard PPD.

Future Directions

Humans are remarkably adapted to solving bilateral social dilemmas. We have focused on recent work that tries to use deep reinforcement learning to give artificial agents this capability. We have shown that amTFT and CCC can maintain cooperation and avoid exploitation in Markov games. In addition we have discussed the training of these strategies and shown that it requires no more than modified self-play. We now highlight important future directions.

The first is game theoretic. We have discussed a conditionally cooperative strategy that uses the intentions behind an action (amTFT) and one purely uses the consequences (CCC). In the real world intentions are generally only partially observed (either because actions are only partially observed or because modeling their future consequences is difficult)



PPD

Strategy	SelfMatch	Safety	IncentC
π^C	0	-18.4	-12.3
π^D	-5.9	0	-18.4
CCC	0	-4.6	3.3
amTFT	-1.6	-5.2	2.6

Risky PPD

Strategy	SelfMatch	Safety	IncentC
π^C	-0.7	-23.6	-12.8
π^D	-5.8	0	-22.6
CCC	-0.2	-12.2	-5.7
amTFT	-3.6	-3.1	2.5

Figure 2: In the PPD both amTFT and CCC agents can be trained from raw pixels. Cooperators can again be exploited by defectors and conditionally cooperative strategies can be both safe and incentivize cooperation. In the non-stochastic version CCC does as well as amTFT but in the stochastic version CCC can be exploited in finite time games while amTFT cannot.

while consequences can sometimes be poor diagnostics for intentions (because of stochasticity). Thus, an important future direction is to construct strategies that combine these two signals.

The second has to do with non-degeneracy of cooperative strategies. The technical conditions for amTFT and CCC to work require the cooperative strategies satisfy a form of exchangeability - that is, given two sets of cooperative policies any re-combination of them leads to the same outcomes. If cooperative policies are not exchangeable we will have both a social dilemma ('should we cooperate?') and a coordination ('in which way should we cooperate?') problem. This is strongly related to work on focal points as well as choosing equilibria in coordination games (Schelling 1980; Peysakhovich and Lerer 2017b). Solving this problem, e.g. via introducing communication, is an important avenue for future work. See Kleiman-Weiner et al. for a more in depth discussion.

The third is algorithmic. Any conditionally cooperative strategy needs access to the cooperative strategy and a 'threat' strategy. In the surveyed papers we used modified self-play to find these strategies. However, to the best of our knowledge there are no guarantees that even if such strategies exist that standard self-play will find them. In addition, self-play can have stability issues in multi-agent systems as the environment from the perspective of a single agent becomes non-stationary due to the fact that other agents are learning (Foerster et al. 2017b; 2017a; Lowe et al. 2017). Finally, in some situations it can be difficult to find the joint payoff maximizing cooperative policy. Dealing with each of these issues is an important step in scaling these ideas to new environments.

The final issue has to do with human psychology. Here we have focused on implementing strategies that achieve socially optimal payoffs (that is, maximizing the sum of payoffs). However, if we are interested in agents that interact with humans this may not be enough. Human social preferences are more complex than this and the kinds of allocations that humans find fair vary greatly among cultures and contexts – sometimes it is fair for one person to get a lot more than the other and other times it is not (Roth et al. 1991; Henrich et al. 2001; Herrmann, Thöni, and Gächter 2008; List 2007). Perceptions of fairness greatly influence behavior and in particular humans are often willing to pay costs to retaliate against an unfair partner (Camerer and Thaler 1995; Fehr and Gächter 2002; Ouss and Peysakhovich 2015). Thus, if an artificial agent tries to behave according to an efficient but unfair policy, it may find itself stuck in π^D even though a better outcome was possible. Understanding social preferences in context is thus an important question to answer if we seek to construct systems which lead to good outcomes (Crandall et al. 2017; Shirado and Christakis 2017; Hauser et al. 2014).

References

Axelrod, R. M. 2006. *The evolution of cooperation: revised edition*. Basic books.

Brown, N.; Ganzfried, S.; and Sandholm, T. 2015. Hierarchical abstraction, distributed equilibrium computation, and post-processing, with application to a champion no-limit texas hold'em agent. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, 7–15. International Foundation for Autonomous Agents and Multiagent Systems.

Brown, G. W. 1951. Iterative solution of games by fictitious play. *Activity analysis of production and allocation* 13(1):374–376.

Camerer, C., and Thaler, R. H. 1995. Anomalies: Ultimatums, dictators and manners. *The Journal of Economic Perspectives* 9(2):209–219.

Crandall, J. W.; Oudah, M.; Ishowo-Oloko, F.; Abdallah, S.; Bonnefon, J.-F.; Cebrian, M.; Shariff, A.; Goodrich, M. A.; Rahwan, I.; et al. 2017. Cooperating with machines. *arXiv preprint arXiv:1703.06207*.

De Cote, E. M., and Littman, M. L. 2012. A polynomial-time nash equilibrium algorithm for repeated stochastic games. *arXiv preprint arXiv:1206.3277*.

Dutta, P. K. 1995. A folk theorem for stochastic games. *Journal of Economic Theory* 66(1):1–32.

Fehr, E., and Gächter, S. 2002. Altruistic punishment in humans. *Nature* 415(6868):137–140.

Foerster, J.; Farquhar, G.; Afouras, T.; Nardelli, N.; and Whiteson, S. 2017a. Counterfactual multi-agent policy gradients. *arXiv preprint arXiv:1705.08926*.

Foerster, J.; Nardelli, N.; Farquhar, G.; Torr, P.; Kohli, P.; Whiteson, S.; et al. 2017b. Stabilising experience replay for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1702.08887*.

Foerster, J. N.; Chen, R. Y.; Al-Shedivat, M.; Whiteson, S.; Abbeel, P.; and Mordatch, I. 2017c. Learning with opponent-learning awareness. *arXiv preprint arXiv:1709.04326*.

Fudenberg, D., and Levine, D. K. 1998. *The theory of learning in games*, volume 2. MIT press.

Fudenberg, D., and Maskin, E. 1986. The folk theorem in repeated games with discounting or with incomplete information. *Econometrica: Journal of the Econometric Society* 533–554.

Hauser, O. P.; Rand, D. G.; Peysakhovich, A.; and Nowak, M. A. 2014. Cooperating with the future. *Nature* 511(7508):220–223.

Henrich, J.; Boyd, R.; Bowles, S.; Camerer, C.; Fehr, E.; Gintis, H.; and McElreath, R. 2001. In search of homo economicus: behavioral experiments in 15 small-scale societies. *The American Economic Review* 91(2):73–78.

Herrmann, B.; Thöni, C.; and Gächter, S. 2008. Antisocial punishment across societies. *Science* 319(5868):1362–1367.

Kleiman-Weiner, M.; Ho, M. K.; Austerweil, J. L.; Michael L, L.; and Tenenbaum, J. B. 2016. Coordinate to cooperate or compete: abstract goals and joint intentions in social interaction. In *Proceedings of the 38th Annual Conference of the Cognitive Science Society*.

Leibo, J. Z.; Zambaldi, V.; Lanctot, M.; Marecki, J.; and Graepel, T. 2017. Multi-agent reinforcement learning in sequential social dilemmas. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, 464–473. International Foundation for Autonomous Agents and Multiagent Systems.

Lerer, A., and Peysakhovich, A. 2017. Maintaining cooperation in complex social dilemmas using deep reinforcement learning. *arXiv preprint arXiv:1707.01068*.

List, J. A. 2007. On the interpretation of giving in dictator games. *Journal of Political economy* 115(3):482–493.

Littman, M. L., and Stone, P. 2005. A polynomial-time nash equilibrium algorithm for repeated games. *Decision Support Systems* 39(1):55–66.

Littman, M. L. 2001. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, 322–328.

Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; and Mordatch, I. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540):529–533.

Nowak, M., and Sigmund, K. 1993. A strategy of win-stay, lose-shift that outperforms tit-for-tat in the prisoner's dilemma game. *Nature* 364(6432):56.

Ouss, A., and Peysakhovich, A. 2015. When punishment doesn't pay: 'cold glow' and decisions to punish. *Journal of Law and Economics* 58(3).

Peysakhovich, A., and Lerer, A. 2017a. Consequentialist conditional cooperation in social dilemmas with imperfect information. *arXiv preprint arXiv:1710.06975*.

Peysakhovich, A., and Lerer, A. 2017b. Prosocial learning agents solve generalized stag hunts better than selfish ones. *arXiv preprint arXiv:1709.02865*.

Roth, A. E.; Prasnikar, V.; Okuno-Fujiwara, M.; and Zamir, S. 1991. Bargaining and market behavior in jerusalem, ljubljana, pittsburgh, and tokyo: An experimental study. *The American Economic Review* 1068–1095.

Sandholm, T. W., and Crites, R. H. 1996. Multiagent reinforcement learning in the iterated prisoner's dilemma. *Biosystems* 37(1-2):147–166.

Schelling, T. C. 1980. *The strategy of conflict*. Harvard university press.

Shapley, L. S. 1953. Stochastic games. *Proceedings of the national academy of sciences* 39(10):1095–1100.

Shirado, H., and Christakis, N. A. 2017. Locally noisy autonomous agents improve global human coordination in network experiments. *Nature* 545(7654):370–374.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. 2016. Mastering the game of go with deep neural networks and tree search. *Nature* 529(7587):484–489.

Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. 2017. Mastering the game of go without human knowledge. *Nature* 550(7676):354–359.

Tampuu, A.; Matiisen, T.; Kodelja, D.; Kuzovkin, I.; Korjus, K.; Aru, J.; Aru, J.; and Vicente, R. 2017. Multiagent cooperation and competition with deep reinforcement learning. *PloS one* 12(4):e0172395.

Tesauro, G. 1995. Temporal difference learning and td-gammon. *Communications of the ACM* 38(3):58–68.